

II. AMENDMENTS TO THE SPECIFICATION:

Please make the following amendments to the specification:

On page 7, para. 18, please amend as follows:

As used herein, the term “prerequisite” is intended to refer to any type of resource that a native application might need to run/operate properly (e.g., as intended) on a client device. Examples of prerequisites include, among other things, other native applications, databases, packages, services, etc. As known, under the OSGi, a “package” is similar to a JAVA package and a “service” is a certain type of interface.

On pages 8-9, para. 0021, please amend as follows:

In any event, assume that native application 18A is to be loaded on client device 14. Under the present invention, prerequisite system 20 will first determine/identify the prerequisites for native application 18A. This can be accomplished by accessing a stored reference based on the identity and version of native application 18, or based on a detailed analysis thereof by prerequisite system 20. Regardless, once the prerequisites are known, corresponding dependency information will be generated. The dependency information typically indicates the one or more prerequisites on which native application 18 depends for proper operation on client device 14. For example, if native application 18A requires another native application 18B (e.g., native application “Y” version 2.0) to properly run on client device 14, dependency information will be generated that specifies this information. Once the dependency information is determined/generated, it will be packaged along with native application 18A within OSGi bundle 16A. Once packaged, OSGi bundle 16A will be registered with server ~~20~~ 12. Registration typically includes storing the identity of native application 18A along with the dependency

information in a registry maintained in local memory, cache or the like.

On page 17, para. 0035, please amend as follows:

In this example, assume that all prerequisites have been resolved and that only OSGi bundles 16A-B need to be loaded on client device 14. In this event, export system 56 of bundle loading system 54 will install OSGi bundles 16A-B (having native applications 18A-B therein) within OSGi environment 22 of client device 14. In installing the bundles 16A-B in this manner, export system 56 could pass an instruction to client device 14 specifying the precise order in which native applications 18A-B should be installed. In any event, deployment system 58 will thereafter deploy ~~to the~~ the native code contained within the OSGi bundles 16A-B to native environment 24. Once deployed, removal system 60 may optionally remove native applications 18A-B from within OSGi bundles 16A-B. At this point, native applications 18A-B are deployed in native environment 24, while OSGi bundles 16A-B are deployed within OSGi environment.

On pages 17-18, para. 0036, please amend as follows:

[0036] Referring now to Fig. 3, a method flow diagram 100 according to the present invention is shown. As depicted, first step S1 is to identify the bundle containing the native application to be loaded on a client device. Second step S2 is to determine whether the native application has any prerequisites. If not, the native application is packaged within an OSGi bundle and loaded on the client device in step S7. If, however, the native application has prerequisites, the native application is ~~packages~~ packaged within an OSGi bundle along with corresponding dependency information in step S3. As indicated above, the dependency information specifies any prerequisite(s) on which the native application depends for proper operation on the client device. Fourth step S4 is to poll the client device to determine if the client device has the at least one

other prerequisite. If it is determined in step S5, that the client device does not lack any of the prerequisites, the OSGi bundle containing the native application and the dependency information is loaded on the client device in step S7. If, however, the client device was determined in step S5 to be lacking any of the prerequisite(s), the server will attempt locate one or more other registered OSGi bundles that provide the missing prerequisites in step S6. If none can be found, the process ends in step S8. If, however, such OSGi bundles are found, the process will be repeated recursively from step S2 for such bundles. Once all prerequisites have been resolved, all necessary OSGi bundles will be loaded on the client device in step S7.